Programming and Data Structure

Indranil Sen Gupta

Department of Computer Science & Engg.
Indian Institute of Technology
Kharagpur

Some General Announcements

About the Course

- Will be conducted with a L-T-P rating of 3-0-0.
- Laboratory with a L-T-P of 0-1-3.
  - Grading will be separate.
- Tutorial classes (one hour per week) will be conducted along with the laboratory.
- Evaluation in the theory course:
  - Mid-semester 30%
  - End-semester 50%
  - Two class tests and attendance 20%

Course Materials

- The slides for the lectures will be made available on the web (in PDF form).

  `http://144.16.192.60/~isg/PDS`

- All important announcements will be put up on the web page.
- Hard copies of the slides will be distributed.
  - Few copies distributed during the class.
  - One copy kept in Ramakrishna Xerox centre.

ATTENDANCE IN THE CLASSES IS MANDATORY

Students having poor attendance will be penalized in terms of the final grade.

Any student with less than 80% attendance would be deregistered from the course, and debarred from appearing in the examinations.

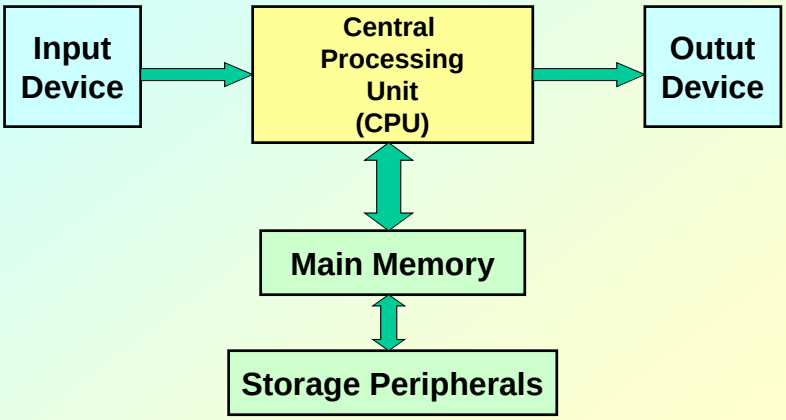
Text/Reference Books & Notes

1. Programming with C (Second Edition)
   B.S. Gottfried, Schaum's Outline Series, Tata McGraw-Hill, 2006.
2. Programming in ANSI C (Second Edition)
   E. Balagurusamy, Tata McGraw-Hill, New Delhi, 1992.
3. Data structures
   S. Lipschutz, Schaum's Outline Series, Tata McGraw-Hill, 2006.
4. Data structures using C and C++ (Second Edition)
   Y. Langsam, M.J. Augenstein, A.M. Tanenbaum, Prentice-Hall of India.
5. http://144.16.192.60/~pds/notes/

Introduction

What is a Computer?

It is a machine which can accept data, process them, and output results.



- **CPU**
  - All computations take place here in order for the computer to perform a designated task.
  - It has a number of registers which temporarily store data and programs (instructions).
  - It has circuitry to carry out arithmetic and logic operations, take decisions, etc.
  - It retrieves instructions from the memory (fetch), interprets (decode) them, and performs the requested operation (execute).

- **Main Memory**
  - Uses semiconductor technology.
  - Memory sizes in the range of 512 Mbytes to 4 Gbytes are typical today.
  - Some measures to be remembered
    - 1 K (kilo) = 2^{10} (= 1024)
    - 1 M (mega) = 2^{20} (= one million approx.)
    - 1 G (giga) = 2^{30} (= one billion approx.)

- Input Device
  - Keyboard, Mouse, Scanner, Touchpad
- Output Device
  - Monitor, Printer
- Storage Peripherals
  - Magnetic Disks: hard disk, floppy disk
    - Allows direct (semi-random) access
  - Optical Disks: CDROM, CD-RW, DVD, BlueRay
    - Allows direct (semi-random) access
  - Flash Memory: pen drives
    - Allows direct access
  - Magnetic Tape: DAT
    - Only sequential access

Typical Configuration of a PC

- CPU: Pentium IV, 2.8 GHz
- Main Memory: 2 GB
- Hard Disk: 300 GB
- Floppy Disk: Not present
- CDROM: DVD combo-drive
- Input Device: Keyboard, Mouse
- Output Device: 17” color monitor
- Ports: USB, Firewire, Ethernet

How does a computer work?

- **Stored program concept.**
  - Main difference from a calculator.
- **What is a program?**
  - Set of instructions for carrying out a specific task.
- **Where are programs stored?**
  - In secondary memory, when first created.
  - Brought into main memory, during execution.

Number System :: The Basics

- We are accustomed to using the so-called ***decimal number system***.
  - Ten digits :: 0,1,2,3,4,5,6,7,8,9
  - Every digit position has a weight which is a power of 10.
- **Example**:

$234 = 2 \times 10^2 + 3 \times 10^1 + 4 \times 10^0$

$250.67 = 2 \times 10^2 + 5 \times 10^1 + 0 \times 10^0 + 6 \times 10^{-1} + 7 \times 10^{-2}$

Contd.

- A digital computer is built out of tiny electronic switches.
  - From the viewpoint of ease of manufacturing and reliability, such switches can be in one of two states, ON and OFF.
  - A switch can represent a digit in the so-called *binary number system*, 0 and 1.
- A computer works based on the binary number system.

- Binary number system
  - Two digits :: 0 and 1
  - Every digit position has a weight which is a power of 2.
- Example:

$1110 = 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$

$= 14$ (in decimal)

Concept of Bits and Bytes

- **Bit**
  - A single binary digit (0 or 1).
- **Nibble**
  - A collection of four bits (say, 0110).
- **Byte**
  - A collection of eight bits (say, 01000111).
- **Word**
  - Depends on the computer.
  - Typically 4 or 8 bytes (that is, 32 or 64 bits).

Contd.

- **An k-digit decimal number**
  - Can express unsigned integers in the range
    0 to $10^k - 1$.
    - For k=3, from 0 to 999.
- **An k-bit binary number**
  - Can express unsigned integers in the range
    0 to $2^k - 1$.
    - For k=8, from 0 to 255.
    - For k=10, from 0 to 1023.

Classification of Software

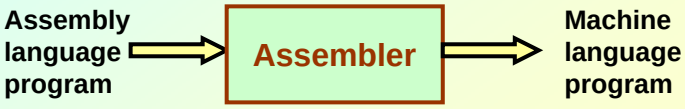
- Two categories:
  1. Application Software
     - Used to solve a particular problem.
     - Editor, financial accounting, weather forecasting, mathematical toolbox, etc.
  2. System Software
     - Helps in running other programs.
     - Compiler, operating system, etc.

Computer Languages

- Machine Language
  - Expressed in binary.
    - 10110100 may mean ADD, 01100101 may mean SUB, etc.
  - Directly understood by the computer.
  - Not portable; varies from one machine type to another.
    - Program written for one type of machine will not run on another type of machine.
  - Difficult to use in writing programs.

Contd.

- **Assembly Language**
  - Mnemonic form of machine language.
  - Easier to use as compared to machine language.
    - For example, use “ADD” instead of “10110100”.
  - Not portable (like machine language).
  - Requires a translator program called ***assembler***.



Contd.

- **Assembly language is also difficult to use in writing programs.**
  - Requires many instructions to solve a problem.
- **Example: Find the average of three numbers.**

```
MOV   A,X    ; A = X
ADD   A,Y    ; A = A + Y
ADD   A,Z    ; A = A + Z
DIV   A,3    ; A = A / 3
MOV   RES,A  ; RES = A
```

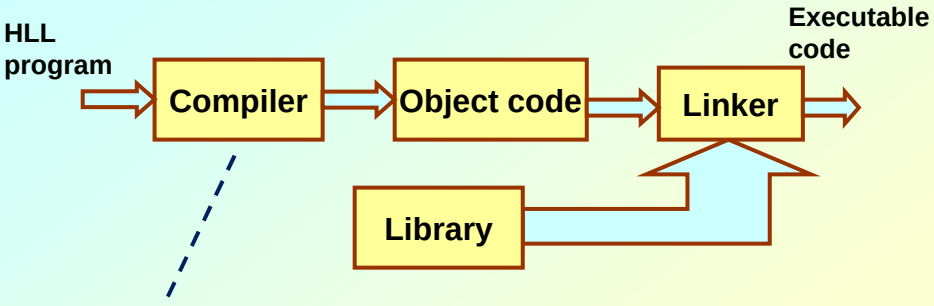
In C,

RES = (X + Y + Z) / 3

High-Level Language

- Machine language and assembly language are called low-level languages.
  - They are closer to the machine.
  - Difficult to use.
- High-level languages are easier to use.
  - They are closer to the programmer.
  - Examples:
    - Fortran, C, C++, Java.
  - Requires an elaborate process of translation.
    - Using a software called *compiler*.
  - They are portable across platforms.

Contd.



gcc compiler will be used in the lab classes

Operating Systems

- Makes the computer easy to use.
  - Basically the computer is very difficult to use.
  - Understands only machine language.
- Operating systems makes the task of the users easier.
- Categories of operating systems:
  - Single user
  - Multi user (Time sharing, Multitasking, Real time)

Contd.

- Popular operating systems:
  - DOS: single-user
  - Windows 2000/XP: single-user multitasking
  - Unix: multi-user
  - Linux: a free version of Unix
- The laboratory class will be based on Sun OS (a version of UNIX).

Contd.

- **Question:**
  - How many users can work on the same computer?
- **Computers connected in a network.**
- **Many users may work on a computer.**
  - Over the network.
  - At the same time.
  - CPU and other resources are shared among the different programs.
    - Called time sharing.
    - One program executes at a time.

The Laboratory Environment